n – 1702

On Multiple Description Streaming with Content Delivery Networks

John Apostolopoulos, Tina Wong, Wai-tian Tan, Susie Wee
Streaming Media Systems Group
Hewlett-Packard Laboratories, Palo Alto, CA

Abstract— **CDNs have been widely used to provide low latency, scalability, fault tolerance, and load balancing for the delivery of web content and more recently streaming media. We propose a system that improves the performance of streaming media CDNs by exploiting the path diversity provided by existing CDN infrastructure. Path diversity is provided by the different network paths that exist between a client and its nearby edge servers; and multiple description (MD) coding is coupled with this path diversity to provide resilience to losses. In our system, MD coding is used to code a media stream into multiple complementary descriptions, which are distributed across the edge servers in the CDN. When a client requests a media stream, it is directed to multiple nearby servers which host complementary descriptions. These servers simultaneously stream these complementary descriptions to the client over different network paths.**

This paper provides distortion models for MDC video and conventional video. We use these models to select the optimal pair of servers with complementary descriptions for each client while accounting for path lengths and path jointness and disjointness. We also use these models to evaluate the performance of MD streaming over CDNs in a number of real and generated network topologies. Our results show that distortion reduction by about 20 to 40% can be realized even when the underlying CDN is not designed with MDC streaming in mind. Also, for certain topologies, MDC requires about 50% fewer CDN servers than conventional streaming techniques to achieve the same distortion at the clients.

I. Introduction

CONTENT delivery networks (CDNs) were developed to overcome performance problems, such as network congestion and server overload, that arise when many users access popular content. CDNs improve end-user performance by caching popular content on edge servers located closer to users. This provides a number of advantages. First, it helps prevent server overload, since the replicated content can be delivered to users from edge servers. Furthermore, since content is delivered from the closest edge server and not from the origin server, the content is sent over a shorter network path, thus reducing the request response time, the probability of packet loss, and the total network resource usage. While CDNs were originally intended for static web content, recently, they have been applied to the delivery of streaming media as well.

Streaming media is characterized by data that has a strict delay constraint. This delay constraint makes streaming media very sensitive to packet loss and network outages. For example, when receiving a streaming media session, data that arrives late is useless. Not only does streaming media suffer from the same problems associated with static content delivery, it also presents additional challenges due to the real-time nature of the content. Conventional approaches for dealing with packet loss for static data, such as retransmissions, may not be possible in a streaming context. Thus, additional mechanisms are needed to provide streaming media delivery over packet networks.

Of the various techniques to improve streaming media quality, a method of *multiple description coding (MDC)* with *path diversity* was proposed in [1]. MDC codes a media stream into two (or more) complementary descriptions. These descriptions have the property that if either description is received it can be used to decode baseline quality video, and both descriptions can be used to decode improved quality video. This is in contrast to conventional video coders (e.g. MPEG-1/2/4, H.261/3, Microsoft's and Real Networks's proprietary coders), which produce a single stream that does not have these MD properties; we refer to these methods as *single description coding (SDC)*.

MDC combines particularly well with path diversity, in which the different descriptions are explicitly sent over different routes to a client. Path diversity exploits the fact that while any network link may suffer from packet loss, there is a much smaller chance that two network paths simultaneously suffer from losses. In other words, losses on the two paths are likely to be uncorrelated. MDC combined with path diversity is beneficial for delay-sensitive, real-time applications such as streaming media, where data losses, especially consecutive ones, are highly disruptive to the application. In prior work [1], path diversity was achieved using either a relay infrastructure or source-based routing.

In this work, we achieve error resilient media streaming by using MDC and leveraging CDN infrastructure to provide path diversity. We use MDC to code a media stream into multiple descriptions, and distribute copies of these descriptions across surrogates in the CDN. When a client requests a media stream, it is directed to multiple nearby surrogates which host complementary descriptions of the stream. The client simultaneously receives the different descriptions through different network paths from the different surrogates. That is, we leverage the existing CDN infrastructure to achieve path diversity between multiple surrogates and the client. In this way, disruption in streaming media occurs only in the less likely case when simultaneous losses afflict both paths. This architecture also reaps the benefits associated with CDNs, such as reduced response time to clients, load balancing across servers, robustness to network and server failures, and scalability to number of clients.

This paper continues in Section II by describing how CDNs can be used to achieve path diversity. Section III discusses architecture design issues that arise in using MDC in CDNs. Section IV characterizes the performance of MDC when used with path diversity. Section V presents simulation results on MD-CDN performance for various network topologies. Section VI mentions additional related work. Section VII concludes with a brief summary.

II. Path Diversity in CDN

Diversity schemes, such as frequency, time, and spatial diversity, have been widely employed to improve system reliability in wireless communications [2]. In wired networks, only time diversity or interleaving is typically exploited due to the lack of infrastructure support for path diversity. However, the benefits of path diversity can be significant due to the potentially highly variable nature of the quality of each individual path [3], and

the often failure in identifying the single best path [4]. IP source routing is one possible mechanism to achieve path diversity but is not widely supported. The advent of the CDN provides a new platform under which path diversity can be realized without resorting to explicit path-diversity mechanisms [1], [5]. By virtue of having the original content replicated at multiple geographically or topologically separated surrogates, a CDN provides a client multiple paths of different characteristics to access the same content.

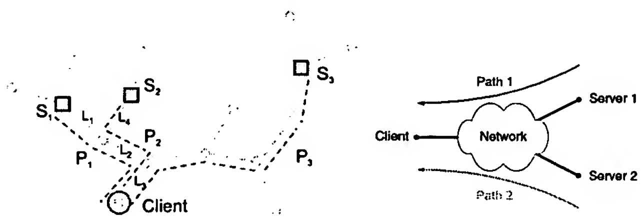


Fig. 1. Exploiting path diversity of a CDN (left) and abstraction of the two-path path diversity between a client and two surrogates within a CDN (right)

Fig. 1 illustrates how path diversity offered by a CDN is typically exploited to provide fault tolerance when only a single path between a client and a surrogate is used. Normally, a client indicated by the circle communicates with the closest surrogate S_1 via path P_1. In the case when link L_1 goes down or S_1 is overloaded, the client can be redirected to an alternate surrogate S_2 and obtain the same content from S_2 via path P_2. Similarly, if link L_2 goes down or S_2 is overloaded, the client can again be redirected to an alternate surrogate S_3 and obtain the same content from S_3 through a longer path P_3. Ideally, the best achievable reception for a single path is attained when the client instantaneously *switch* to the best path. However, due to the lack of *a priori* information to facilitate instantaneous switching, reactive switching is used. As a result, the applicability of the scheme is restricted to more persistent network impairments, such as network outages, and is largely ineffective against transient network losses.

We propose an additional way to exploit path diversity provided by CDNs by enabling simultaneous communication between a client and multiple surrogates. For instance, the client in Fig. 1 can obtain half of the content from S_1 using path P_1, and the other half from S_2 via P_2. When link L_1 goes down, the client can reactively switch to using S_2 and S_3 through paths P_2 and P_3, respectively, achieving fault tolerance. One key advantage of simultaneously using multiple paths is the reduction in the probability of simultaneous, correlated loss in all paths, regardless of the loss characteristics of individual links. Whether such a feature can be translated into benefits for video applications depends on our ability to exploit it. In Section IV, we will describe MDC and its relevant characteristics for transmission using multiple paths.

Of course, the use of multiple paths does not guarantee independence of the paths. Generally, parts of the paths may be disjoint while other parts may overlap. For instance, when paths P_1 and P_2 in Fig. 1 are used, the links L_2 and L_3 are shared while links L_1 and L_4 are not. When losses occur in either link L_1 or L_4, only one path is affected. On the other hand, if losses occur in either link L_2 or L_3, both paths are affected. Thus, the advantage of having multiple paths depends on the characteristics of the "joint" part of the paths. If most of the losses occur in the joint part of the paths, there is little advantage in using multiple paths. Conversely, if the "joint" part has relatively little losses, then the benefit of using multiple paths is enhanced. If paths P_2 and P_3 of Fig. 1 are used instead of paths P_1 and P_2, the number of joint links is decreased from two to one, which may be preferrable despite the fact that P_3 is longer than P_1. The impact of having losses in the joint and disjoint parts of each path are examined in the context of MDC and path diversity in Section IV. We also examine the joint/disjoint path charcteristics for real and generated network topologies in Section V.

III. MD-CDN Architecture Design

This section discusses the architectural design issues that arise when using CDN's for delivering MD coded content. We refer to such a system as a Multiple Description CDN (MD-CDN). Some of these issues are also found in a traditional streaming CDN, which we refer to as Single Description CDN (SD-CDN), but require alternative solutions to optimize for the MD case. In the case of MD streaming within an existing CDN infrastructure, the design issues that arise include (1) how to distribute the MD streams across the existing surrogates, a process which we refer to as *MD distribution across surrogates*, and (2) how to select for each client multiple surrogates with complementary descriptions, which we refer to as *MD surrogate selection*. In the case of deploying a MD-CDN from scratch, the design issue of optimal *MD surrogate placement* also arises.

A. MD Distribution ("Coloring") Across Surrogates

Previous work on server and CDN surrogate placements [6], [7] focus on placing mirrors or replicas, in which the assumption is that complete content is stored at each chosen server. For MD-CDN, this assumption is invalid because a unit of content (e.g. a movie) is divided into multiple complementary descriptions that are spread across a number of surrogates. To be specific, for MD with two descriptions, in general each surrogate may host 0, 1, or both descriptions. An important special case is when each surrogate hosts one description, which leads to the notion of *coloring* where we assign to each surrogate a particular color corresponding to a unique description. The goal of this coloring problem is to color the surrogates so that a complete set of descriptions (e.g. both descriptions for MD with two descriptions) are close to every client. While this notion of coloring is useful, it is also unnecessarily constraining since in general each surrogate may host both or neither descriptions.

B. MD Surrogate Selection

A number of selection algorithms are possible with varying deployment complexity and MDC-biased performance optimization. Current CDN request (re-)direction mechanisms, such as DNS request routing, assume a single path when selecting a surrogate for a client. Therefore, they do not consider properties of multiple paths, such as disjointness. Nonetheless, these mechanisms can be applied in a MD-CDN setting, by simply choosing the best N surrogates with N complementary descriptions, where best is determined by, e.g., shortest path. More sophisticated algorithms optimized for MD-CDN that require additional network and systems support are also possible, and would improve the performance of MD-CDN. In Section V we present two algorithms that take into account specific properties of MD and path diversity in the surrogate selection process.

We evaluated the performance of a combination of surrogate coloring and selection algorithms in Section V.

C. MD Surrogate Placement

Surrogate placement is the problem of finding the best locations to deploy surrogates to optimize client performance. This is also called the "facility location problem" or the "p-center problem", and a number of graph theoretic approximation algorithms have been proposed. Previous work [6], [7] in the SD-CDN case places surrogates in order to minimize distances from surrogates to clients when servicing requests. For SD-CDN researchers, the main goal is to come up with practical approaches that are deployable in the current Internet infrastructure. However, to optimally place surrogates for a MD-CDN is more complex than for a SD-CDN because there are two potentially opposing objectives: minimize distance from clients to surrogates, and maximize path disjointness between multiple surrogates and each client.

Although the ideal MD surrogate placement would account for both path distance and diversity between clients and surrogates, current CDN and data center infrastructures are designed to optimize only for the former but not the latter. For example, Akamai [8] already has more than 10,000 surrogates located at the edge of the Internet to reduce client access time to surrogates; Digital Island has data centers at a few well-connected places in the world. Therefore, it may not be practical to require a MD-CDN to work off a completely different set of surrogates. Rather, for ease of deployment and economic reasons, it may be highly beneficial for the MD-CDN to leverage existing CDN and data center infrastructures to deliver descriptions from multiple surrogates to a client. We show in Sections V that MD performs well in a SD-optimized CDN, where servers are located either at the edge or in the core of the network.

IV. Multiple Description Coding and Path Diversity

MD coding has been shown to provide improved performance in networks with path diversity [1]. This section characterizes the performance of MDC in the context of the type of path diversity that can be achieved in a CDN.

A. Multiple Description Video Coding

Multiple Description Coding (MDC) refers to a form of compression where a signal is coded into a number of separate bitstreams, where the multiple bitstreams are referred to as multiple descriptions (MD). These multiple descriptions provide two important properties. First, each description can be decoded independently to give a usable reproduction of the original signal. Second, the multiple descriptions contain complementary information so that the quality of the decoded signal improves with the number of descriptions that are correctly received.

An important point is that each description or MD bitstream is independent of each other and is typically of roughly equal importance. This is in contrast to conventional layered or scalable schemes. Layered or scalable approaches essentially prioritize data and thereby support intelligent discarding of the data (the enhancement data can be lost or discarded while still maintaining usable video). However the base-layer bitstream is critically important – if it is lost then the other bitstream(s) are useless. MD coding overcomes this problem by allowing useful reproduction of the signal when *any* description is received, and with increasing quality when more descriptions are received.

In the context of path diversity where each path simultaneously carries a different description, the properties of MDC suggest that a usable quality is maintained whenever any description is correctly received. Since using multiple paths reduces the probability of having simultaneous losses in all the paths, a scheme in which MDC is used with multiple paths improves the chance of receiving at least a usable quality of video.

A number of MD video coding algorithms have recently been developed, which provide different tradeoffs in terms of compression performance and error resilience [9], [10], [11], [12]. In this paper we base our work on the MD video coder presented in [12], [1]. Some important characteristics of this coder are: (1) high compression efficiency, achieving MDC properties with only slightly higher total bit rate than conventional SD compression schemes, (2) ability to use correctly received descriptions to repair corrupted descriptions over time, (3) ability to successfully operate over paths that support different or unbalanced bit rates [13], and (4) standard compatibility, with this MD coder being a *standard-compatible enhancement* to MPEG-4 Version 2 (with NEWPRED) and H.263 Version 2 (with RPS). A consequence of (4) is that any MPEG-4 V2 decoder can decode the MD bitstream while an enhanced decoder designed to perform state recovery as presented in [12] can provide improved error recovery. In addition, this form of MD video coding contains conventional (SD) coding as a special case, thereby enabling an encoder to adapt its processing between SD and MD based on the current communication characteristics.

As discussed before, a general MD coder is designed to operate assuming at least one description is correctly received. This assumption can be quite restrictive, i.e. over the duration of a video session both descriptions will generally be partially received and partially corrupted. One notable benefit of our selected MD coder is that it allows repair of corrupted descriptions using uncorrupted descriptions so that usable quality can be maintained even when there are losses in all descriptions, as long as the losses do not simultaneously afflict all descriptions.

B. Loss Characteristics of SD and MD Video Streams

This section examines the MD and conventional SD performance for streaming video test sequences over a lossy packet network. Specifically, the effect of single and burst losses are examined in the SD and MD contexts; and in the MD context, the effect of losses in one and both network paths are examined.

Two test sequences were used. Foreman is a head-and-shoulders type sequence similar to a videoconferencing application (144 × 176 pixels/frame at 30 frames/sec) while Bus is a more complicated sequence similar to a conventional movie (240 × 352 pixels/frame at 30 frames/sec). The MD coder coded each sequence into two descriptions, corresponding to the even and odd frames. The SD and MD video coding algorithms were based on the MPEG-4/H.263-like coder described in Section IV-A. To make an appropriate comparison, the sequences were coded with MD and SD at the same constant video quality and the same total bitrate (bits/sec). Each coder uses a different approach for error-resilience: MD via the MD properties, while SD devotes extra bits for additional intraframe coding to enable it to recover faster from losses. For simplicity, we assume that each packet loss results in the loss of an entire frame. This assumption is appropriate for the Foreman sequence, where an entire predictively coded frame fits within a single packet, however it is a worst-case assumption for the Bus sequence, where a predictively coded frame typically requires about 5 packets. Details

of the specific comparisons are given in [1].

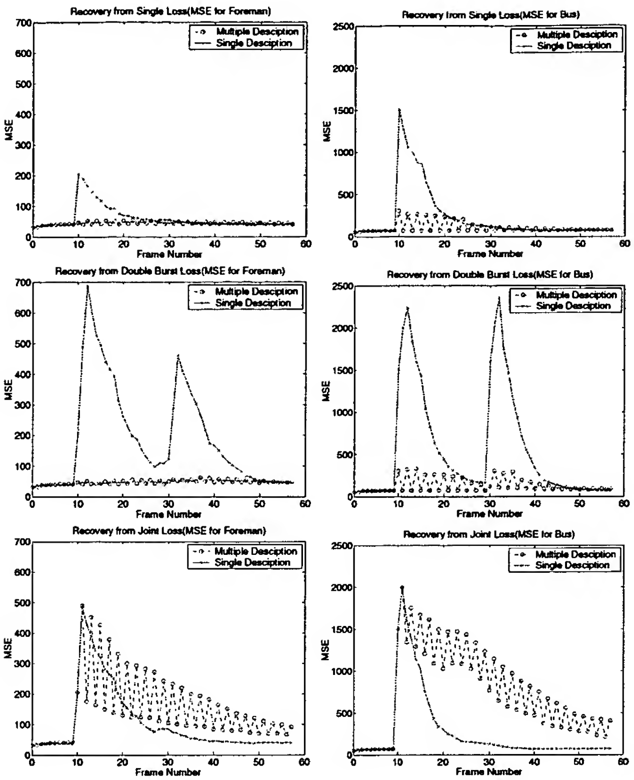


Fig. 2. Recovered SD and MD video quality for the Foreman (left) and Bus (right) sequences in single and burst losses in one and both channels.

Figure 2 illustrates the performance for MD and SD video coding under three types of losses: (1) single loss corresponding to the loss of a single entire frame (frame 10), (2) two burst losses of 100 ms duration, spaced apart by 2/3 sec, which corresponds to the loss of three frames in two locations spaced apart by 2/3 sec (starting at frame 10 and frame 30 and afflicting both MD streams), and (3) simultaneous losses in both streams. Specifically, in case (2) three frames are lost in the even sequence starting at frame 10, and three frames are lost in the odd sequence starting at frame 31 (2/3 sec later). Distortion is measured in terms of mean-squared error (MSE), and a lower distortion (MSE) indicates better quality.

We make the following conclusions about SD and MD performance in the face of packet loss. For a single loss (top row), the SD error is characterized by an initial jump in distortion and a gradual recovery. The MD error is characterized by a very small jump in the corresponding affected even or odd subsequence. The smaller jump in distortion for MD is because the correctly received neighboring frames are used in this form of MD coding to perform state recovery to accurately recover the lost frame.

For a burst loss (middle row), the SD error is characterized by a large jump for each consecutive packet loss and a gradual recovery. For a burst loss in one MD path, the MD error is similar to that of a *single* loss; consecutive losses do not result in accumulated distortion because the state recovery at the decoder can recover using correctly received neighboring frames. For both SD and MD, losses spaced far enough apart behave as independent losses.

For a burst loss (bottom row), the SD error is once again characterized by a large jump for each consecutive error and a gradual recovery. For simultaneous losses in both MD paths, the error is characterized by a jump in distortion for the even and odd subsequences, and each gradually recovers. Note that the MD rate of recovery is slower than that of SD because MD coding uses less intraframe coding (given the same total bit rate constraint).

MD coding is more resilient to single losses and burst losses than SD coding as long as the losses afflict only one channel at a time. In this case, the correctly received channel can be used to recover the corrupted channel with state recovery techniques as described in [12], [1]. Because of this bootstrap off of the correctly received channel, MD coding is largely immune to the duration of loss in one channel. In the case of simultaneous errors affecting both channels, SD recovers more quickly because of the extra intraframe coding that can be used.

We quantify the distortion for SD through 7 distortion parameters: distortion for (1) no loss, (2) loss of one frame, (3) recovery after loss of one frame, (4) loss of a second frame, (5) recovery after loss of second frame, (6) loss of a third frame, (7) recovery after loss of a third frame. We assume that the distortion saturates for burst loss length larger than 3. The distortion for MD is quantified with 5 distortion parameters (assuming balanced or symmetric MD): distortion of one description for (1) no loss, (2) loss of one frame (affecting only one description), (3) recovery after loss, (4) simultaneous loss of both descriptions, (5) recovery after simultaneous loss. Note that it is unnecessary to account for burst length in MD since it is largely immune to (independent of) the length of the loss as long as the loss afflicts only a single description at any point in time.

C. Modeling Loss Characteristics of SD and MD Streams

This section describes models for comparing SD and MD video delivery quality as a function of path characteristics and losses. The distortion metric is the mean-square error (MSE) in the reconstructed video at the decoder. As discussed in the prior section, a number of different types of losses afflict conventional SD and MD video in important and different ways and therefore must be accounted for. These events include isolated packet loss, burst loss as well as the specific length of the burst, and in addition for MD whether the loss afflicts only a single description at any point in time or simultaneously afflicts both descriptions. To distinguish between these different loss events requires a model that can express burst loss and furthermore can capture and distinguish between the losses that occur on joint and disjoint links. In the following, we present models of the end-to-end loss processes for single and multiple paths, and corresponding distortion models that map loss events for SD and MD to actual distortions. Specifically, the distortion models capture the important loss events described above, and the model for the end-to-end loss process for two-path path diversity accounts for both joint and disjoint links.

To model and characterize the performance of MD and SD delivery over a simulated network, we introduce two simplifying assumptions. First, given a network with a number of links, we assume that the burst loss behavior of each link can be modeled by a two-state Gilbert model parameterized by transition probabilities $\{p_0, q_0\}$, where p_0 is the probability of going from no loss (0) to loss (1) and q_0 is the probability of going from loss (1) to no loss (0). The Gilbert model is widely used to model bursty traffic for its simplicity and mathematical tractability. While prior work modeled end-to-end packet loss across a single path

in the Internet [14], [15], we propose single link models which are then used to develop end-to-end loss models. Second, we also assume that each link can be modeled as independent.

A path is modeled as the concatenation of a number of bursty single links. When each bursty single link in a path of N links is modeled as a Gilbert model, it can be shown that the end-to-end probability of loss and loss runlength can be captured by a single Gilbert model parameterized by a different set of transition probabilities $\{p_{total}, q_{total}\}$.

For SD delivery over a single path of N_{single} links, the end-to-end loss characteristics is modeled by a two-state Gilbert model whose parameters depend on the number of links (path length N_{single}) and the parameters for each link. This model expresses the loss process for the path but not the distortion when video is transmitted over that path. One distortion model for SD video over a single path is the 4-state model shown in Figure 3, where the states denote the number of consecutive losses in the immediate past. The distortion for bursts of length longer than 3 is approximated by that of a length 3 burst. Note that the transition probabilities in the distortion model are determined by the parameters of the Gilbert model for the path only, while the distortion associated with the state transitions is a function of the video source only. The 7 SD distortion parameters quantify the distortion for each of the transitions. Given this distortion model, the average distortion for a particular source and path can be easily computed using the stationary distribution of the states.

MD with two descriptions and two paths from a client to two servers is much more complex than SD over a single path, as the client and servers can be connected through a wide range of different topologies, and different links may be joint (shared) by both paths while other links may be disjoint (not shared). However, it can be shown [16] that to capture the desired end-to-end characteristics, we do not need to distinguish based on the specific topology. Instead we can summarize the path diversity to a given client simply in terms of three subpaths and the parameters corresponding to the lengths of these subpaths: (1) disjoint links along the first path, (2) joint links along the first and second paths, and (3) disjoint links along the second path. Therefore, the loss process for two-path path diversity from a MD-CDN to a given client can be expressed by the triplet $\{N_{Disjoint\text{-}1}, N_{Joint}, N_{Disjoint\text{-}2}\}$, where the total number of links in the first path is $N_{Disjoint\text{-}1} + N_{Joint}$, and the total number of links in the second path is $N_{Joint} + N_{Disjoint\text{-}2}$. In conclusion, we do not need to distinguish based on the specific topology, and instead can summarize the path diversity via three subpaths, each modeled by a two-state Gilbert model which corresponds to the concatenation of multiple (bursty) single links of that subpath. While this system may be modeled with an 8-state model, the Cartesian product of the three two-state Gilbert sub-paths, the need to distinguish the losses that afflict each description in the joint subpath and the dependencies between these losses requires a 4-state model for the joint subpath. In addition, the packet rate (packets/sec) for each joint or disjoint link must be appropriately accounted for in terms of its Gilbert parameters. In summary, the loss process for two-path path diversity can be modeled with a 16-state model and a corresponding 16×16 state transition matrix that expresses the transition probabilities from one time instant to the next.

To model MD application-level quality, we map the above model, which expresses the loss process for two-path path diversity, to an application-level model which expresses the end-to-end distortion behavior of both descriptions sent over their respective paths. It is clear from Figure 2 that the distortion for MD video, unlike for SD video, depends critically on whether loss afflicts both descriptions at the same time, rather than the burst loss length on any single description. Therefore, an appropriate model that captures the distortion behavior of an MD source is the 4-state model in Figure 4, which expresses at each point in time whether both descriptions are correctly received (state 00), one description is correctly received and one description is afflicted by losses (states 01 and 10) and both descriptions are simultaneously afflicted by losses (state 11). Specifically, the 16-state path diversity model is mapped to the 4-state application-layer (source) model, where each of the 16 possible transitions corresponds to a different loss event and a different distortion in the reconstructed video. Each of the 16 transition probabilities corresponds to the sum of a subset of the 256 transition probabilities in the 16×16 state transition matrix of the two-path path diversity loss process. The expected MD distortion is computed based on the 4-state model where the distortion for each transition is quantified by a different combination of the 5 MD distortion parameters. Specifically, the total expected distortion is given by the sum of the products of the steady state probability for each state times the transition probability out of that state times the distortion that results from that transition. It is useful to note that the proposed loss model for path diversity may also be useful for other applications not related to MD coding. Similarly, while the specifics of this MD distortion model were chosen to accurately represent our MD coder, other forms of MD coding may be analyzed using a very similar model.

For convenience of simulation, we assume each link is identical and parameterized by Gilbert parameters $\{p_0, q_0\}$. Therefore, given a topology, for our simulations we construct the Gilbert parameters for each link to produce end-to-end characteristics similar to those measured in the Internet.

To summarize, assuming all links are identical, the expected distortion (MSE) for SD is given by $D_{SD}(N_{single}, p_0, q_0)$ and that for MD by $D_{MD}(N_{Disjoint\text{-}1}, N_{Joint}, N_{Disjoint\text{-}2}, p_0, q_0)$.

As an example, Figure 5 illustrates the relative performance of MD and SD when a client is connected via a symmetric "Y" ($N_{Disjoint\text{-}1} = N_{Disjoint\text{-}2}$) and we vary the fraction of the total number of links that are joint and disjoint. Specifically, the total length of each path is 8 links, and the number of joint links is varied from 0 to 8 and the number of disjoint links therefore varies from 8 to 0. For this plot we assumed $\{p_0, q_0\} = \{.0052, .8\}$ for each link, where the p_0 corresponds to 5 % end-to-end average packet loss for 8 links, and $q_0 = .8$ corresponds to the longest average burst length (assuming 30 msec sampling) that we are aware of in the literature [14], [15].

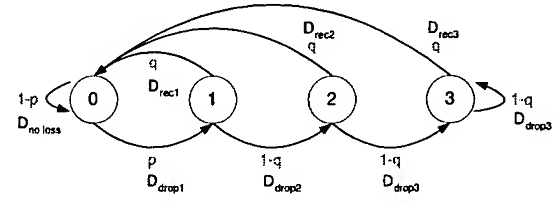


Fig. 3. Expected SD video quality is estimated from this model, where the four states identify the burst length and where the transition probabilities are labeled as well as the distortions that result for those transitions.

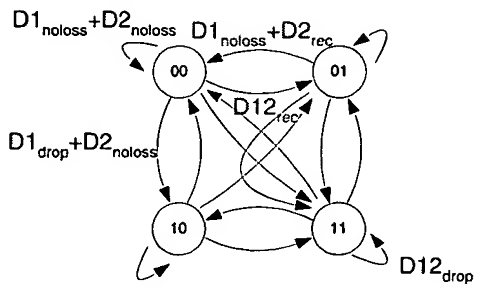


Fig. 4. Expected MD video quality is estimated from this model, where the four states identify at any instant in time whether any of the two descriptions are currently afflicted by losses. Each of the 16 different transition arcs corresponds to a different distortion (only 4 are labeled).

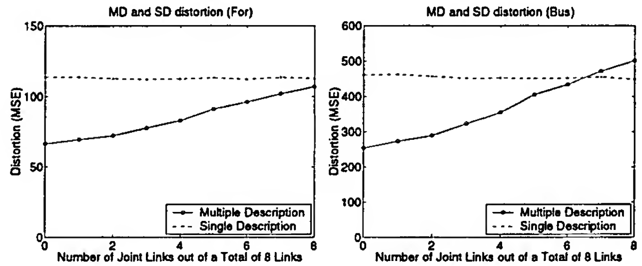


Fig. 5. MD versus SD distortion for a symmetric "Y", as we vary the number of joint links given that the total number of links is 8. MD provides less distortion than SD for all cases of Foreman (right) and almost all cases for Bus (left) except where the paths are almost completely joint.

V. Simulation Experiments

We conducted simulation experiments to study how an MD-CDN behaves under various conditions. The goal of the experiments is to examine two questions. First, we investigated whether and how much a MD-CDN is able to yield performance improvement over SD-CDN, while leveraging only existing server infrastructures such as CDN and Internet Data Centers (IDC). Second, we studied the sensitivity of MD-CDN to MD-optimized algorithms which require information not available in SD-CDN. We also examined the path diversity characteristics of a number of real and generated network topologies.

A. Methodology

In our simulation experiments, we placed servers on both generated and existing network topologies, and then colored them with different MD streams. A pair of servers hosting complementary descriptions is selected for each client request. For both SD-CDN and MD-CDN, the collective performance across all clients for each network topology is evaluated. We varied a number of parameters in our experiments– topology, placement, coloring and selection algorithms– which are discussed next.

A.1 Topology

To determine how MD-CDN fares in different networks, we examined in our experiments a number of different topologies which are listed with their characteristics in Table I. The AT&T and UUNet ISP backbone graphs are available from CAIDA [17]. The AS graph, from NLANR [18], corresponds to connectivity among Internet autonomous systems (AS) where each node in the graph represents an AS. We also examined generated topologies created by the BRITE [19] topology generator

Name	Type	Date	# Nodes	# Edges
AT&T	ISP	2000	87	195
UUNet	ISP	2001	113	1078
AS	Inter-AS	1999	4830	9078
BRITE-h	Generated	NA	1000	1987
BRITE-f	Generated	NA	1000	1997

TABLE I
TOPOLOGIES.

from Boston University. BRITE models incremental growth and preferential connectivity in networks [20], [21], which are possible causes for power-laws observed in Internet topologies [22]. Using BRITE, we created a two-level hierarchical and a one-level flat topology that models the Internet.

A.2 MD Surrogate Placement Algorithms

We used the following placement algorithms to place servers on a subset of nodes in a topology:

• *Edge*: To emulate surrogate placement in a CDN, we place servers at the edge of a topology, which we define as nodes with degree of two to three. If there are more candidate edge nodes than desired number of servers, a random tie breaker is used.
• *Core*: To emulate data center placement at the most connected part of a network, we place servers at the core of a topology, which we define as nodes with the highest degree. Again, a random tie breaker is used to select among multiple nodes with the same degree.
• *IDC*: For some ISP graphs, the location of Internet Data Centers (IDC) are available. The IDC locations of the AT&T IP backbone are available [23], which we use to place servers for the AT&T topology in our experiments. This emulates "hotspot" placement where client population is most concentrated.

We used these simple placement algorithms in the experiments to examine whether MD-CDN can leverage existing server infrastructures that are not optimized for MD. All the algorithms listed above are biased towards SD, such that the distance from servers to clients (Edge) or from servers to servers (Core and IDC) are minimized. While the ideal case is to use a real surrogate location graph from a CDN company, such information is proprietary and not available.

A.3 MD Distribution Across Surrogates Algorithms

Given a placement of servers, an important question is how to distribute the MD descriptions across the surrogates. In general each surrogate may host 0, 1, or both descriptions. In the following we use the conceptually useful, though suboptimal, notion of coloring, where each surrogate is assigned a single description. To compare SD-CDN and MD-CDN performance, we instrumented the following coloring algorithms to create one SD and two MD scenarios:

• *SD*: The *SD* algorithm randomly selects half of the available servers, and places SD at each server. This is to model SD-CDN in which each server stores the full content. We also use SD as the baseline algorithm to compare with MD-based approaches.
• *MD-half*: On the same half of the servers selected by *SD*, the *MD-half* algorithm places both descriptions at each server. As explained in Section IV, we encode the Bus and Foreman video sequences into two descriptions such that the resulting total bit rate equals that of the SD stream. Hence, the *MD-half* algorithm imposes the constraints that SD and MD use the same servers,

and also use the same total amount of (1) storage in the infrastructure and (2) bandwidth to the clients.

• *MD-all*: The *MD-all* algorithm randomly places one of the two descriptions at each and every available server, with the condition that at least one server is assigned to each color. Here, we remove the constraint that SD and MD use the same servers, however the total storage used in the infrastructure remains the same for SD and MD, as well as bandwidth to the clients.

The motivation for using these coloring algorithms in the simulation experiments is to examine the hypothesis that placing only one description on each server, but using twice as many servers, could provide improved path diversity than the conventional approach where each server would store the complete video or both descriptions.

A.4 MD Surrogate Selection Algorithms

Given placement and coloring of servers with the above algorithms, the surrogate selection problem addresses how to select for each client the optimal pair of multiple surrogates with complementary descriptions while accounting for path lengths and path jointness and disjointness. Conventional CDN selection assume a single surrogate over a single path, and select the best surrogate based on, for example, shortest path. This may be extended to an MD-CDN by selecting the two surrogates with shortest paths, however this does not consider the jointness or disjointness of the paths. More sophisticated algorithms that take into account specific properties of MD and path diversity in the surrogate selection process can provide improved performance. To evaluate these benefits we instrumented the following selection algorithms in our simulation:

• *Shortest Path (SP)*: Pick the two closest servers (with different descriptions) to the client. We measure closeness by hop counts. If more than one server has the same shortest path distance, a tie breaker is chosen randomly.

• *Heuristic*: For each pair of servers S_i, S_j with complementary descriptions, we calculate a score using the equation, $\frac{(p_i+p_j)}{2} + N_{Joint_{i,j}}$, where p_i is the path length in hop counts from S_i to the client (i.e. $p_i = N_{Joint_{i,j}} + N_{Disjoint_i}$), and p_j from S_j to the client, and where $N_{Joint_{i,j}}$ is the path length of the joint portion of the two paths. This heuristic algorithm aims to minimize joint path and total path lengths between a client and its two servers. Note that we assume the two servers are unique ($i \neq j$), i.e. even if a server is close to the client and has both descriptions, it will not stream both MD streams to the client.

• *Distortion*: For each pair of servers with complementary descriptions, we calculate the expected distortion for a client using the method described in Section IV-C. The pair of servers with the lowest estimated distortion is then chosen for the client.

The above algorithms are ordered according to increasing deployment complexity and MD-biased optimization. *SP* is analogous to current CDN request direction using DNS, and does not consider path characteristics such as disjointness. *Heuristic* requires knowledge of paths between each client-server pair, thus may demand either dynamic network support or static topology snapshots at the selection algorithm. *Distortion* requires distortion parameters for each stream, in addition to knowledge of server-client paths. However given this knowledge, our analytical models enable us to determine the optimal pair of surrogates for each client, in terms of minimizing the expected distortion. Note that this selection problem is particularly important, as it must be solved every time any client requests any content. We instrumented these algorithms to evaluate the benefits to MD-CDN of information that is not necessary in SD-CDN. This gives us intuition on the incremental deployment issues of MD-CDN.

Model	Topology	% Servers	Placement	Coloring	Selection
IDC	AT&T	10	IDC	*	*
	UUNet	10	Core	*	*
CDN	AS	1	Edge	*	*
	BRITE-h	1	Edge	*	*
	BRITE-f	1	Edge	*	*

TABLE II

SUMMARY OF EXPERIMENTS WE CONDUCTED. "*" DENOTES ALL ALGORITHMS ARE EVALUATED.

A.5 Packet Loss Model

We used the Gilbert loss model developed in Section IV to simulate packet losses in our experiments. We fixed $q = 0.8$ which corresponds to an expected burst loss length of 1.25; studies [14], [15] have shown that consecutive losses (loss run-lengths) are short and rarely last more than four packets, and this q corresponds to the longest average burst length measurement that we are aware of. We chose p to yield a moderate end-to-end loss rate of 5% for an average path length of five or eight hops (depending on topology).

Table II summarizes the experiments we conducted.

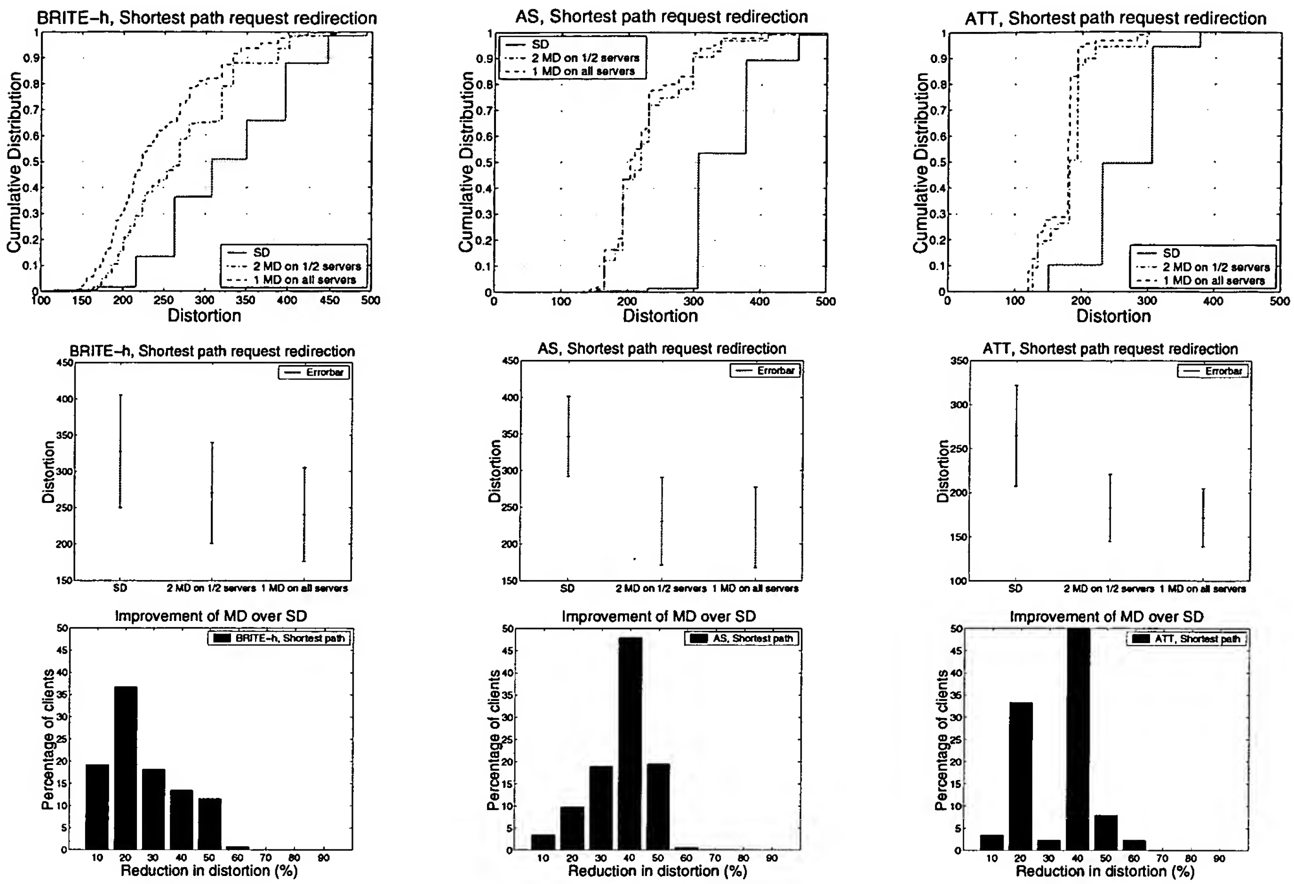
B. Simulation Results

B.1 MD-CDN Performance in SD-biased Environments

The first question we asked in our simulation experiments is whether and how much MD-CDN yield improvements over SD-CDN, while leveraging only existing CDN and IDC infrastructures. In particular, given servers that are placed to minimize distance to clients, or servers that are located in the core of the network, is there enough path diversity in such environments that MD can utilize to reduce distortions at the clients. We also assumed in this part of the experiment only simple network support is available. To direct client requests to servers, we simply find the two servers with the shortest paths to the client. As discussed in the methodology section, we evaluated one SD scenario, and two MD approaches where in the first both descriptions are stored in half of the available servers (MD-half), and in the second one description is stored in every server (MD-all).

Figure 6 are results for the BRITE-h, AS and ATT graphs. Because of limited space, we do not show plots for all topologies we examined. For each topology, we calculated the following for each of the SD and MD scenarios. First, cumulative distribution of distortion for clients. This shows us the general performance of MD-CDN over SD-CDN–specifically whether MD-CDN yields lower distortion for the clients. Second, we calculated the mean and standard deviation of the distortions over all clients in a topology. Third, we drew a histogram of the reduction in distortion achieved at each client if MD-CDN (MD-all) is used instead of SD-CDN. This part of the experiment is based on the Bus video sequence which is a complicated sequence and for which MD and SD have relatively close performance. For the Foreman sequence MD provides significantly better performance than SD, and we do not include those results.

From the cumulative distribution plots of Figure 6, we see that in general MD-CDN outperforms SD-CDN, even when the



(a) BRITE-h (b) AS (c) ATT

Fig. 6. Top row: Cumulative distribution of distortion at the clients for SD and the two MD scenarios for BRITE-h (left column), AS (middle column), and ATT (right column). Middle row: Average distortion and its standard deviation for each scenario. Bottom row: Histograms of the percentage reduction in distortion when MD is used over SD. We observe that both MD-half and MD-all outperform SD in terms of lower expected distortion at the clients in all three networks.

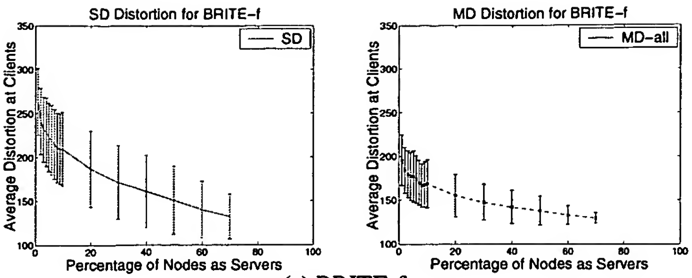
placement, coloring and selection algorithms are optimized towards SD. To state specific numbers, for the BRITE-h topology, about 13% of the clients experience distortion of 250 or less if SD is used to deliver the Bus sequence, but this is true for more than 40% to 60% of the clients for both MD scenarios. We observe similar results for the rest of the topologies.

The mean and standard deviation plots and the histogram plots give us a better idea of how much MD-CDN reduces distortion over SD-CDN. We see that for all topologies evaluated, the average distortion for both MD scenarios are lower than SD. For example, in the AT&T ISP backbone topology, the average is approximately between 160 to 180 for MD, whereas SD clients would experience an average distortion of over 250. We also observe that, if we utilize all available servers but only store one description on each (MD-all), the performance is slightly better than if we store both descriptions in only half of the servers (MD-half). We found that the average client-server path length is shorter in MD-all than MD-half, which follows intuition because more servers are available in MD-all to service the same number of clients, thus MD-all is able to provide lower distortion numbers. This reinforces our proposal of spreading MD over servers, instead of storing all descriptions for a video sequence on each server.

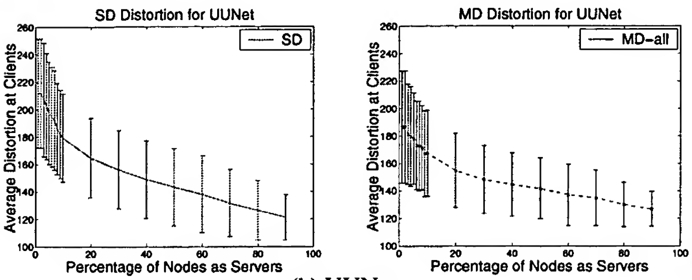
To dig a bit deeper, the histogram plots quantify the improvement of MD-CDN over SD-CDN. We compare the MD-all scenario to SD since we found that MD-all is slightly better than MD-half. The reduction in distortion for each client is calculated as $\frac{|MSE_{SD}-MSE_{MD}|}{MSE_{SD}}$, where MSE_{SD} is the distortion for SD, and MSE_{MD} is the distortion for MD. The histogram plots show that, for certain topologies such as AS and AT&T, most clients see a 40% distortion reduction. We suspect that the different results arise from the different characteristics of the topologies. For example, AS in particular is well connected, with a few node degrees of over 100, thus yielding short but diverse routers between nodes.

To compare and contrast MD-CDN and SD-CDN from another viewpoint, we examine the number of servers necessary to achieve an average distortion at the clients for both schemes. Figures 7 illustrates the reduction in distortion at the clients as we increase the percentage of nodes acting as servers. We observe that MD streaming requires fewer servers than SD streaming to achieve the same average distortion. For example, Figure 7(a) shows that in the BRITE-f topology, SD-CDN would need 50% of the nodes acting as servers to achieve an average distortion of 150, whereas it only takes MD-CDN about 30%. We see similar results for other topologies. To get an average distortion of 170 in the AS topology, SD streaming would need approximately 40% of its nodes acting as servers, whereas MD about 15%. Also, the variance of distortion at the clients is smaller in the case of MD streaming, as illustrated by the errorbars.

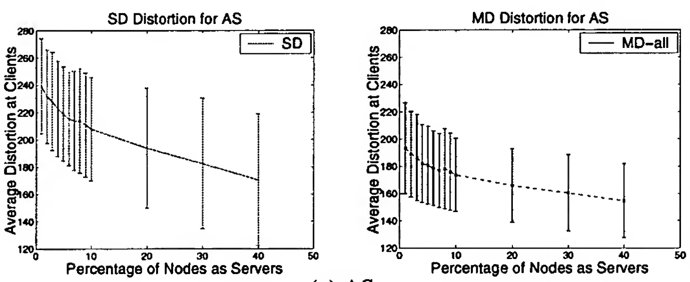
To summarize, this part of the experiment shows that MD streaming performs better than SD streaming in existing CDN



(a) BRITE-f



(b) UUNet



(c) AS

Fig. 7. Average distortion at the clients versus percentage of nodes acting as servers: MD (right) requires fewer servers than SD (left) to achieve a certain distortion, and also provides a lower variance in distortion over the clients.

and IDC infrastructures–without using MD-optimized server placement, coloring and selection that consider path characteristics. MD streaming also requires fewer servers than SD streaming to achieve the same average distortion at clients.

B.2 Benefits of Joint/Disjoint Link Knowledge

We have shown above that MD-CDN outperforms SD-CDN with SD-biased placement, coloring and selection algorithms. In the following, we investigate the additional improvement MD-CDN provides over SD-CDN given joint/disjoint path information (which is not necessary in SD-CDN). We assume that all links are identical. Specifically, we compare the *Distortion* selection algorithm to direct a client to appropriate servers, which assumes knowledge of joint and disjoint links between each client-server pair, to the *Shortest Path (SP)* algorithm which only requires knowledge of path length. Figure 8 compares the reduction in distortion for each algorithm. These specific test conditions favor shortest path selection (for identical links, the problem largely reduces to minimizing path lengths) and knowledge of joint/disjoint links provides marginal additional gain over simply selecting the closest two *distinct* servers. However, in the more typical case where different links have different loss characteristics, the use of this information may provide significant improvement over a selection based solely on path length.

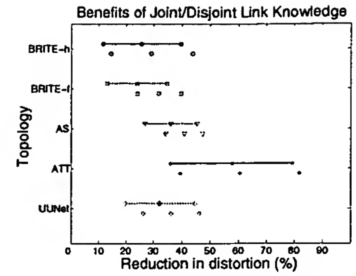


Fig. 8. Improvements of MD-CDN over SD-CDN when joint/disjoint link information is (1) known, and (2) not known. For each topology, there are two lines: the dotted line below denotes the average and one standard deviation above and below the mean of reduction in distortion when joint/disjoint link information is known, and the solid line above is for when it is not known. All links within a topology are assumed identical.

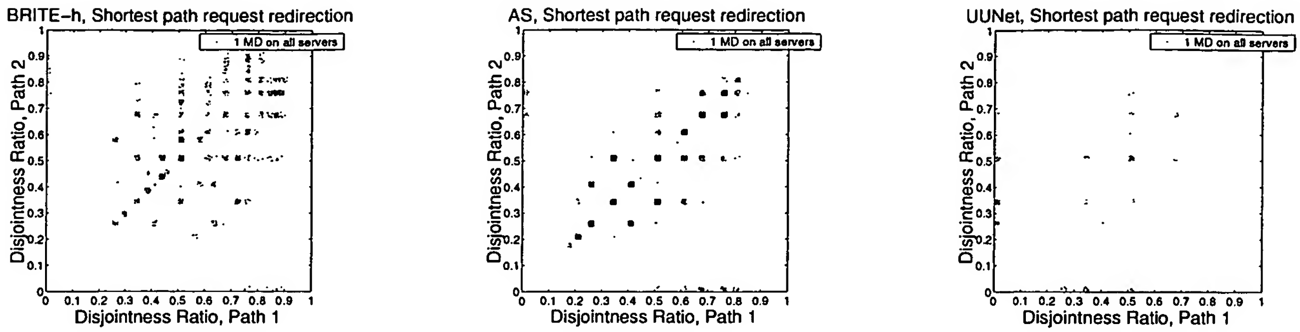
B.3 Correlation of Disjointness Ratios

It is also interesting to investigate the disjointness of paths between a client and its MD servers in the various topologies. Figure 9 illustrates this correlation for BRITE-h, AS and UUNet graphs. The x-axis denotes disjointness ratio on the first path, and y-axis the second path, where disjointness ratio is given by $\frac{N_{Disjoint}}{p}$ where $N_{Disjoint}$ is the disjoint path length and p is the total path length. In other words, the larger the disjointness ratio, the more disjoint is the path. A dot in the scatterplot located at $x = d1, y = d2$ means there is a client with disjointness ratio $d1$ on the first path and $d2$ on the second path. For each topology, we calculated the disjointness ratio for each client to its two *MD-all* colored surrogates, selected with the *Distortion* algorithm. We make a number of observations. For all topologies, there are only a few dots in the upper left-hand and the lower right-hand regions in the scatterplots. A dot in one of these regions signifies asymmetric disjointness ratios, which may arise when a client is located very close to (or co-located with) a server, thus the disjointness ratio on that path is very small (or close to zero). In general, however, most of the dots lie on or around the diagonal, thus disjointness is more symmetrical than asymmetrical.

VI. Additional Related Work

This paper is based on applying MD coding and path diversity in the context of CDN. The idea of using diversity over packet networks in not new, however it has received relatively little attention, where Dispersity Routing by Maxemchuk [24] is one of the first works, and [25] is a more recent example. The approach of this paper is to leverage the CDN surrogate infrastructure to provide multiple paths, without requiring explicit path diversity support from the network.

In prior work [1], [13], MD and path diversity was shown to provide improved performance for point-to-point communication over lossy packet networks, when diversity was achieved using either a relay infrastructure or source-based routing. The idea of using path diversity for point-to-point video/image applications is also proposed by [26], [27] for mobile multihop radio environments, where an MD image coder is used to code each frame into two descriptions based on a checker-board pattern, and recent extensions to video over ad-hoc wireless networks is considered in [28]. In the recent work [29] it is further shown that path diversity can improve latency and loss characteristics for real-time voice communication over the Internet by exploiting the different delay variations along different paths.



(a) BRITE-h (b) AS (c) UUNet

Fig. 9. Correlation of disjointness ratios on client-server paths. Each dot in the scatterplot represents a client in the topology. We add small "jitter" to the dots to help distinguish them clearly. We observe that disjointness of the two client-server paths are somewhat symmetrical, and the upper left-hand and lower right-hand regions which denote severe disjointness asymmetry are largely unoccupied.

Digital Fountain [30] applies Tornado codes to achieve reliable data download. Their subsequent work [31] reduces download times by having a client receive a Tornado encoded file from multiple mirror servers. The target application of their approach is bulk data transfer, not real-time video. On the other hand, our paper focus on streaming media with MDC.

Another interesting, though more distant work, is Resilient Overlay Networks (RON) which provide resilience to network failures by using an overlay to re-route around failures [32].

VII. Summary

The combination of multiple description coding and path diversity provide improved error-resilience for streaming media over best-effort networks. In this work, we use CDNs to explicitly provide multiple paths over which to deliver complementary descriptions from different edge servers to a single client. We show how path diversity can be achieved with CDNs and present models for estimating the performance of MD and path diversity. We examine the surrogate coloring and selection problems for MD-CDNs, and propose an algorithm to select the optimal pair of surrogates hosting complementary descriptions for each client. We believe the proposed metric for evaluating MD and path diversity performance for a client-server pair is a key component for designing more sophisticated algorithms for MD surrogate placement and MD distribution across surrogates ("coloring"). We conducted simulation experiments on various topologies and settings, and found that MD streaming performs better than SD in existing infrastructures–without MD-optimized server placement, coloring, or selection algorithms–thereby reducing distortion at the clients for the same number of surrogates, or reducing the required number of surrogates to achieve a desired distortion. In summary, our results show that coupling MD coding with path diversity from a CDN can provide significant performance benefits over a conventional SD-CDN.

References

[1] J.G. Apostolopoulos, "Reliable video communication over lossy packet networks using multiple state encoding and path diversity," *Visual Communications and Image Processing (VCIP)*, January 2001.

[2] H.V. Poor and G.W. Wornell, *Wireless Communications: Signal Processing Perspectives*, Prentice Hall, N.J., 1998.

[3] V. Paxson, "End-to-end internet packet dynamics," *Proc. of the ACM SIGCOMM*, pp. 139–152, Sept. 1997.

[4] A. Collins Savage, S., E. Hoffman, J. Snell, and T. Anderson, "The end-to-end effects of internet path selection," *Proceedings of the ACM SIGCOMM*, October 1999.

[5] J.G. Apostolopoulos and G.W. Wornell, "A system for enabling reliable communication over lossy packet networks via path diversity," *HP Internal Report, To be published externally*, 1999.

[6] L. Qiu, V. Padmanabhan, and G. Voelker, "On the placement of web server replicas," *INFOCOM*, 2001.

[7] S. Jamin, C. Jin, A. Kurc, D. Raz, and Y. Shavitt, "Constrained mirror placement on the internet," in *Proceedings of INFOCOM*.

[8] "The Akamai Web Site," http://www.akamai.com/.

[9] S. Wenger, G. Knorr, J. Ott, and F. Kossentini, "Error resilience support in H.263+," *IEEE Transactions on Circuits and Systems for Video Technology*, pp. 867–877, November 1998.

[10] V. Vaishampayan and S. John, "Interframe balanced-multiple-description video compression," *ICIP*, Oct. 1999.

[11] A.R. Reibman, H. Jafarkhani, Y. Wang, M.T. Orchard, and R. Puri, "Multiple description coding for video using motion compensated prediction," *IEEE Inter. Conf. Image Processing*, October 1999.

[12] J.G. Apostolopoulos, "Error-resilient video compression via multiple state streams," *Proc. International Workshop on Very Low Bitrate Video Coding (VLBV'99)*, pp. 168–171, October 1999.

[13] J.G. Apostolopoulos and S.J. Wee, "Unbalanced multiple description video communication using path diversity," *ICIP*, October 2001.

[14] M. Yajnik, S. Moon, J. Kurose, and D. Towsley, "Measurement and modelling of the temporal dependence in packet loss," *INFOCOM'99*.

[15] J. Wenyu and H. Schulzrinne, "Modeling of packet loss and delay and their effects on real-time multimedia service quality," *NOSSDAV*, 2000.

[16] J.G. Apostolopoulos, W. Tan, S.J. Wee, and G.W. Wornell, "Modeling path diversity for multiple description video communication," *to appear in ICASSP*, May 2002.

[17] "http://www.caida.org/tools/visualization/mapnet," .

[18] "http://moat.nlanr.net/routing/rawdata," .

[19] "http://www.cs.bu.edu/brite," .

[20] A.L. Barabasi and R. Albert, "Emergence of scaling in random networks," *Science*, 1999.

[21] A. Medina, I. Matta, and J. Byers, "On the origin of power-laws in internet topologies," *ACM Computer Communication Review*, 2000.

[22] M. Faloutsos, P. Faloutsos, and C. Faloutsos, "On power-law relationships of the internet topology," *SIGCOMM*, 1999.

[23] "http://www.ipservices.att.com/backbone," .

[24] N.F. Maxemchuk, *Dispersity Routing in Store and Forward Networks*, Ph.D. thesis, University of Pennsylvania, May 1975.

[25] A. Banerjea, "Simulation study of the capacity effects of dispersity routing for fault-tolerant real-time channels," *Computer Communications Review (ACM SIGCOMM'96)*, vol. 26, no. 4, pp. 194–205, October 1996.

[26] N. Gogate and S.S. Panwar, "Supporting video/image applications in a mobile multihop radio environment using route diversity," *Proc. Int. Conf. Communications*, June 1999.

[27] N. Gogate, D. Chung, S.S. Panwar, and Y. Wang, "Supporting image/video applications in a mobile multihop radio environment using route diversity and multiple description coding," *Preprint*.

[28] S. Lin, S. Mao, Y. Wang, and S. Panwar, "A reference picture selection scheme for video transmission over ad-hoc networks using multiple paths," *ICME*, Aug 22-25 2001.

[29] Y.J. Liang, E.G. Steinbach, and B. Girod, "Real-time voice communication over the internet using packet path diversity," *Proc. ACM Multimedia*, Sept/Oct 2001.

[30] J. Byers, M. Luby, M. Mitzenmacher, and A. Rege, "A digital fountain approach to reliable distribution of bulk data," *SIGCOMM*, 1998.

[31] J. Byers, M. Luby, and M. Mitzenmacher, "Accessing multiple mirror sites in parallel: Using tornado codes to speed up downloads," *INFOCOM'99*.

[32] D.G. Andersen, H. Balakrishnan, M.F. Kaashoek, and R. Morris, "The case for resilient overlay networks," *Proc. HotOS VIII*, May 2001.